

WHAT IS CLAIMED IS:

1. A test specification formation supporting apparatus comprising:

5 a test factor classification table forming unit which analyzes class design information of software designed by an object orientation and forms a template of a test factor classification table in which test factors which influence the operation of a method of a test target and conditions of the test factors have been listed; and

10 a test specification forming unit which forms a test specification by combining the conditions described in the test factor classification table completed on the basis of the template of said test factor classification table.

2. An apparatus according to claim 1, wherein:

15 said test factor classification table forming unit comprises a class design information analyzing unit which inputs the class design information of a designated class and analyzes it;

20 a factor classifying unit which extracts test factors including an array of arguments of the method and conditions of the arguments and header information which is used for said test factor classification table from an analysis result of said class design information; and

a classification table output unit which shapes said extracted test factors and header information into a predetermined table format and outputs the template of said test factor classification table; and

25 said test factor specification forming unit comprises a classification table analyzing unit which inputs said test factor classification table and analyzes it;

a test factor combination forming unit which selects one of the conditions of the text factors every test factor from the analysis result of said classification table, forms a combination lest the selected condition is overlapped, and extracts the header information which is used in said test specification; and

a test specification output unit which shapes all of the combinations and the header information formed and extracted by said combination forming unit into a predetermined format and outputs the test specification.

3. An apparatus according to claim 2, wherein when a form of the extracted test factors is a class, the factor classifying unit of said test factor classification table forming unit recursively executes processes for developing attribute values with reference to the corresponding class design information and extracting the developed attribute values as test factors.

4. An apparatus according to claim 3, wherein said test factor classification table forming unit further has a development upper limit number designating unit which designates a development upper limit number of attribute values, and said factor classifying unit does not develop the attribute values which exceed the upper limit number designated by said development upper limit number designating unit.

5. An apparatus according to claim 3, wherein said test factor classification table forming unit further has an unnecessary-development class designating unit which designates a class in which the development

of the attribute values is unnecessary, and said factor classifying unit does not develop the attribute values of the class designated by said unnecessary-development class designating unit upon development of the attribute values.

- 5
6. An apparatus according to claim 2, wherein
- 10 said test factor classification table forming unit further has a method design information analyzing unit which inputs structured class design information regarding processing contents of the designated method and analyzes it,
- 15 said factor classifying unit extracts branching conditions described in the processing contents of said method from a result of said analysis and extracts variables such as arguments and the like constructing said branching conditions and conditions of said variables as test factors, and
- 20 said classification table output unit shapes said extracted test factors and header information into the predetermined table format and outputs the template of said test factor classification table.

- 25
7. An apparatus according to claim 6, wherein said test factor classification table forming unit further has a condition type discriminating unit which discriminates whether said branching condition indicates a normality process or an abnormality process, and a discrimination type indicative of the normality process or the abnormality process is also written in said test factor classification table.

8. An apparatus according to claim 2, wherein said test factor

classification table forming unit further has:

a method contents description sentence analyzing unit which inputs design information described by sentences regarding processing contents of the designated method and analyzes a description of conditions written by a specific notation; and

a condition searching unit which extracts branching conditions described in the processing contents of said method from an analysis result of said design information and searches variables such as arguments and the like constructing said branching conditions and conditions of said variables as test factors,

and said classification table output unit shapes said test factors extracted by said factor classifying unit and said condition searching unit into the predetermined table format and outputs the template of said test factor classification table.

9. An apparatus according to claim 8, wherein said condition searching unit further discriminates whether the branching condition written in said sentences by the specific notation indicates a normality process or an abnormality process, and a discrimination type indicative of the normality process or the abnormality process is also written in said test factor classification table.

10. An apparatus according to claim 9, wherein said test factor classification table forming unit further has a notation designating unit which designates the notation of the conditions which are extracted from the sentences in which the processing contents of the method have been described and the notation of the type indicative of the normality process

or the abnormality process.

11. An apparatus according to claim 2, wherein said test factor classification table forming unit further has an output target selecting unit which selects the method from design information on the basis of a designated method selection reference and allows the test factor classification table as an output target to be formed.

12. An apparatus according to claim 2, wherein said test factor classification table forming unit further has an output target selecting unit which selects the class and the method from design information on the basis of a designated class selection reference and allows the test factor classification table as an output target to be formed.

13. An apparatus according to claim 2, wherein said test factor classification table forming unit further has:

a source code analyzing unit which inputs source codes of a program corresponding to said class design information and analyzes them; and

a source condition searching unit which extracts a condition sentence of the method serving as a target from an analysis result of said source codes, extracts variables used in said condition and the condition sentence as test factors, and adds them into the test factor classification table formed from said class design information.

14. An apparatus according to claim 7 or 9, wherein said test specification forming unit further has a combination evaluating unit

which, when the conditions described in said test factor classification table are combined, designates an upper limit of the number of conditions including the conditions which are classified into the abnormality process.

15. An apparatus according to claim 7 or 9, wherein said test specification forming unit further has a combination evaluating unit which, when the conditions described in said test factor classification table are combined, designates one of a plurality of normality conditions which are classified into the normality process in the same test factor as a representative normality condition and allows said representative normality condition to be combined with the abnormality conditions which are classified into the abnormality process.

16. An apparatus according to claim 7 or 9, wherein said test specification forming unit further has a combination evaluating unit which, when the conditions described in said test factor classification table are combined, restricts the normality conditions which can be combined with respect to the combination of only the normality conditions.

17. A test specification formation supporting method comprising:
a classification table forming step which analyzes class design information of software designed by an object orientation and forms a template of a test factor classification table in which test factors which influence the operation of a method as a test target and conditions of the test factors are listed; and

a test specification forming step which forms a test specification by combining conditions described in said test factor classification table completed on the basis of the template of said test factor classification table.

5

18. A method according to claim 17, wherein:
in said classification table forming step,
the class design information of the designated class is
inputted and analyzed,

10

test factors including an array of arguments of the method
and conditions of the arguments and header information which is used in
said test factor classification table are extracted from an analysis result of
said class design information, and

15

said extracted test factors and header information are shaped
into a predetermined table format, and the template of the test factor
classification table is outputted; and

20

in said test specification forming step,
said test factor classification table is inputted and analyzed,
one of the conditions of the test factors is selected every test
factor from an analysis result of said classification table, a combination is
formed lest it is overlapped, and header information which is used for the
test specification is extracted, and

25

all of the formed combinations and the header information
are shaped into a predetermined format and the test specification is
outputted.

19. A method according to claim 18, wherein in said

classification table forming step, when a form of the extracted test factors is a class, processes for developing attribute values with reference to the corresponding class design information and extracting the developed attribute values as test factors are recursively executed.

20. A method according to claim 19, wherein in said classification table forming step, further, a development upper limit number of attribute values is designated, and the attribute values which exceed the designated development upper limit number are not developed upon development of the attribute values.

21. A method according to claim 19, wherein in said classification table forming step, further, a class in which the development of the attribute values is unnecessary is designated, and the attribute values of the designated class are not developed upon development of the attribute values.

22. A method according to claim 18, wherein
in said classification table forming step, further,
structured class design information regarding processing
contents of the designated method is inputted and analyzed,
branching conditions described in the processing contents of
said method are extracted from a result of said analysis and variables
such as arguments and the like constructing said branching conditions
and conditions of said variables are extracted as test factors, and
said extracted test factors and header information are shaped
into the predetermined table format and the template of said test factor

classification table is outputted.

23. A method according to claim 22, wherein in said classification
table forming step, further, whether said branching condition indicates a
normality process or an abnormality process is discriminated, and a
discrimination type indicative of the normality process or the abnormality
process is also written in said test factor classification table.

24. A method according to claim 18, wherein in said classification
table forming step, further,

design information described by sentences regarding
processing contents of the designated method is inputted and a
description of conditions written by a specific notation is analyzed, and
branching conditions described in the processing contents of
said method are extracted from an analysis result of said design
information and variables such as arguments and the like constructing
said branching conditions and conditions of said variables are searched
as test factors, and

said extracted test factors are shaped into the predetermined
table format and the template of said test factor classification table is
outputted.

25. A method according to claim 24, wherein, further, whether
the branching condition written in said sentences by the specific notation
indicates a normality process or an abnormality process is discriminated,
and a discrimination type indicative of the normality process or the
abnormality process is also written in said test factor classification table.

26. A method according to claim 25, wherein in said classification table forming step, further, the notation of the conditions which are extracted from the sentences in which the processing contents of the method have been described and the notation of the type indicative of the normality process or the abnormality process are designated.

27. A method according to claim 18, wherein in said classification table forming step, further, the method is selected from design information on the basis of a designated method selection reference and the test factor classification table as an output target is formed.

28. A method according to claim 18, wherein in said classification table forming step, the class and the method are selected from design information on the basis of a designated class selection reference and the test factor classification table as an output target is formed.

29. A method according to claim 18, wherein in said classification table forming step, further,

source codes of a program corresponding to said class design information are inputted and analyzed, and

a condition sentence of the method serving as a target is extracted from an analysis result of said source codes, and variables used in said condition sentence and the condition sentence are extracted as test factors and added into the test factor classification table formed from said class design information.

30. A method according to claim 23 or 25, wherein in said test specification forming step, further, when the conditions described in said test factor classification table are combined, an upper limit of the number of conditions including the conditions which are classified into the abnormality process is designated.

31. A method according to claim 23 or 25, wherein in said test specification forming step, further, when the conditions described in said test factor classification table are combined, one of a plurality of normality conditions which are classified into the normality process in the same test factor is designated as a representative normality condition and combined with the abnormality conditions which are classified into the abnormality process.

32. A method according to claim 23 or 25, wherein in said test specification forming step, further, when the conditions described in said test factor classification table are combined, the normality conditions which can be combined are restricted with respect to the combination of only the normality conditions.

33. A test specification formation supporting program which allows a computer to execute:

a classification table forming step which analyzes class design information of software designed by an object orientation and forms a template of a test factor classification table in which test factors which influence the operation of a method as a test target and conditions of the test factors are listed; and

a test specification forming step which forms a test specification by combining conditions described in said test factor classification table completed on the basis of the template of said test factor classification table.

34. A program according to claim 33, wherein:

in said classification table forming step,

the class design information of the designated class is inputted and analyzed,

test factors including an array of arguments of the method and conditions of the arguments and header information which is used in said test factor classification table are extracted from an analysis result of said class design information, and

said extracted test factors and header information are shaped into a predetermined table format, and the template of the test factor classification table is outputted; and

in said test specification forming step,

said test factor classification table is inputted and analyzed,

one of the conditions of the test factors is selected every test factor from an analysis result of said classification table, a combination is formed lest it is overlapped, and header information which is used for a test specification is extracted, and

all of the formed combinations and the header information are shaped into a predetermined format and the test specification is outputted.

35. A computer-readable recording medium which stores a test

specification formation supporting program for allowing a computer to execute:

a classification table forming step which analyzes class design information of software designed by an object orientation and forms a template of a test factor classification table in which test factors which influence the operation of a method as a test target and conditions of the test factors are listed; and

a test specification forming step which forms a test specification by combining conditions described in said test factor classification table completed on the basis of the template of said test factor classification table.

36. A medium according to claim 35, wherein
in said classification table forming step,
the class design information of the designated class is

inputted and analyzed,
test factors including an array of arguments of the method
and conditions of the arguments and header information which is used in
said test factor classification table are extracted from an analysis result of
said class design information, and

said extracted test factors and header information are shaped
into a predetermined table format, and the template of the test factor
classification table is outputted; and

in said test specification forming step,
said test factor classification table is inputted and analyzed,
one of the conditions of the test factors is selected every test
factor from an analysis result of said classification table, a combination is

formed lest it is overlapped, and header information which is used for a test specification is extracted, and

all of the formed combinations and the header information are shaped into a predetermined format and the test specification is outputted.

37. A test factor classification table formation supporting apparatus comprising:

a class design information analyzing unit which inputs class design information of a designated class of software designed by an object orientation and analyzes it;

a factor classifying unit which extracts test factors including an array of arguments of the method and conditions of the test factors and header information which is used for said test factor classification table from an analysis result of said class design information; and

a classification table output unit which shapes said extracted test factors and header information into a predetermined table format and outputs a template of said test factor classification table.

38. A test factor classification table formation supporting method comprising the steps of:

inputting class design information of a designated class of software designed by an object orientation and analyzing it;

extracting test factors including an array of arguments of the method and conditions of the test factors and header information which is used for a test factor classification table from an analysis result of said class design information; and

shaping said extracted test factors and header information into a predetermined table format and outputting a template of said test factor classification table.

5 39. A test factor classification table formation supporting program which allows a computer to execute:

a step which inputs class design information of a designated class of software designed by an object orientation and analyzes it;

10 a step which extracts test factors including an array of arguments of the method and conditions of the test factors and header information which is used for said test factor classification table from an analysis result of said class design information; and

15 a step which shapes said extracted test factors and header information into a predetermined table format and outputs a template of said test factor classification table.

40. A computer-readable recording medium which stores a test specification formation supporting program for allowing a computer to execute:

20 a step which inputs class design information of a designated class of software designed by an object orientation and analyzes it;

a step which extracts test factors including an array of arguments of the method and conditions of the test factors and header information which is used for a test factor classification table from an analysis result of said class design information; and

25 a step which shapes said extracted test factors and header information into a predetermined table format and outputs a template of

said test factor classification table.

10024188 123101